

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A function object for use in creating a mapping in a mapping tool with a graphical user interface between a source object having a source object node and a target object having a target object node, comprising:

a script component having computer-executable instructions for performing a function;

a graphical component associated with the function having an input and an output, the graphical component configured to allow a user to graphically associate the input with a source object node and to associate the output with a target object node in the graphical user interface, the source object node being one of a plurality of hierarchically displayed nodes conforming to a first schema and the target object node being one of a plurality of hierarchically displayed nodes conforming to a second schema; and

an interface component having a globally unique identifier, the interface component provides the script component to a compiler in the mapping tool and provides the graphical component to the graphical user interface.

2. (Previously presented) The function object of claim 1, the interface component further includes a category identifier component.

3. (Previously presented) The function object of claim 1, the interface component further comprises a class identifier component.

4. (Previously presented) The function object of claim 1, the interface component further comprises a function category identifier component associated with the function.

5. (Previously presented) The function object of claim 4, the function category identifier component includes one of string, mathematical, logical, date, conversion, scientific, advanced, and custom.
6. (Previously presented) The function object of claim 1, further comprising a second interface component that allows a user to drag and drop the graphical component in the graphical user interface.
7. (Previously presented) The function object of claim 6, the second interface component is associated with the function object by the mapping tool.
8. (Previously presented) The function object of claim 1, further comprising:
a plurality of script components having computer-executable instructions for performing a plurality of functions; and
a plurality of graphical components individually associated with one of the plurality of functions, individually having an input and an output, and individually configured to allow a user to graphically associate the input with a source object node and to associate the output with a target object node in the graphical user interface, the interface component provides the plurality of script components to a compiler in the mapping tool and provides the plurality of graphical components to the graphical user interface.
9. (Previously presented) The function object of claim 8, the interface component further comprises a plurality of function category identifier components individually associated with the plurality of functions.
10. (Previously presented) The function object of claim 9, at least one of the plurality of function category identifier components includes one of string, mathematical, logical, date, conversion, scientific, advanced, and custom.

11. (Previously presented) The function object of claim 10, the plurality of script components having computer-executable instructions for performing a plurality of functions are in a scripting language.

12. (Previously presented) The function object of claim 11, the scripting language is one of java script, visual basic script, and visual C++.

13. (Previously presented) The function object of claim 1, the script component having computer-executable instructions for performing the function is in a scripting language.

14. (Previously presented) The function object of claim 13, the scripting language is one of java script, visual basic script, and visual C++.

15. (Previously presented) The function object of claim 1, the function object is a COM object, and the interface component is a COM interface.

16. (Currently amended) In a mapping tool with a graphical user interface, a method of creating a mapping between a source object having a source object node and a target object having a target object node, the method comprising:

providing a function object having a script component with computer-executable instructions for performing a function, a graphical component associated with the function and having an input and an output, and an interface component;

displaying the graphical component in the graphical user interface;

graphically associating a source object node with the input using a user interface selection device, the source object node displayed hierarchically amongst a plurality of source object nodes conforming to a first document structure;

graphically associating a target object node with the output using the user interface selection device, the target object node displayed hierarchically amongst a plurality of target object nodes conforming to a second document structure; and

creating a mapping including the computer-executable instructions, the mapping is operative to perform the function according to the source object node and to provide an output value associated with the target object node according to the function.

17. (Previously presented) The method of claim 16, providing the function object further comprises obtaining the function object from a host computer.

18. (Previously presented) The method of claim 17, obtaining the function object from a host computer further comprises obtaining the function object from a DLL file in the host computer.

19. (Previously presented) The method of claim 18, obtaining the function object from a DLL file in the host computer further comprises:

- searching the DLL files in the host computer;
- identifying the function object in the searched DLL files; and
- loading the function object into the mapping tool.

20. (Previously presented) The method of claim 16, providing the function object further comprises obtaining the function object from a global communications network.

21. (Previously presented) The method of claim 20, obtaining the function object from a global communications network further comprises obtaining the function object from an Internet web site.

22. (Previously presented) The method of claim 21, obtaining the function object from an the Internet web site further comprises one of searching the Internet for the function object according to a mapping tool startup script, searching the Internet for the function object according to a user command, and searching a web site provided by the user for the function object.

23. (Previously presented) The method of claim 16, further comprising:
providing a plurality of script components with computer-executable instructions for performing one of a plurality of functions, a plurality of graphical components individually associated with one of the plurality of functions and individually having an input and an output, the interface component is associated with the plurality of script components and the plurality of graphical components; and
displaying at least one of the plurality of graphical components in the graphical user interface.

24. (Previously presented) The method of claim 16, providing the function object further comprises creating a wrapper object in the mapping tool, including the interface component of the function object and a second interface component configured to allow a user to drag and drop the graphical component in the graphical user interface.

25. (Previously presented) The method of claim 16, displaying the graphical component further comprises obtaining the graphical component from a function object source via the interface component.

26. (Previously presented) The method of claim 25, the function object source is a DLL file on a host computer.

27. (Previously presented) The method of claim 25, obtaining the graphical component from a function object source comprises obtaining the graphical component from a DLL file via the Internet.

28. (Previously presented) The method of claim 16, displaying the graphical component further comprises:
displaying a function object palette in the graphical user interface;
displaying the graphical component on the function object palette; and
allowing a user to drag and drop the graphical component from the function object palette to a mapping screen region in the graphical user interface.

29. (Previously presented) The method of claim 28, further comprising:
providing a plurality of script components with computer-executable instructions for performing one of a plurality of functions, a plurality of graphical components individually associated with one of the plurality of functions and individually having an input and an output, the interface component is associated with the plurality of script components and the plurality of graphical components;
displaying a function object palette in the graphical user interface;
displaying at least one of the plurality of graphical components on the function object palette; and
allowing a user to drag and drop the at least one of the plurality of graphical components from the function object palette to a mapping screen region in the graphical user interface.

30. (Previously presented) The method of claim 16, the mapping tool further comprises a compiler component, and creating the mapping further comprises:
invoking the compiler component to generate compiled output code; and
providing the computer-executable instructions from the script component to the compiler via the interface component.

31. (Previously presented) The method of claim 16, the mapping tool further comprises a compiler component, and creating the mapping further comprises:
providing a plurality of script components with computer-executable instructions for performing one of a plurality of functions, a plurality of graphical components individually associated with one of the plurality of functions and individually having an input and an output, the interface component is associated with the plurality of script components and the plurality of graphical components;
displaying a function object palette in the graphical user interface;
displaying at least one of the plurality of graphical components on the function object palette;

allowing a user to drag and drop the at least one of the plurality of graphical components from the function object palette to a mapping screen region in the graphical user interface;

invoking the compiler component to generate compiled output code; and
providing the computer-executable instructions from the plurality of script components to the compiler via the interface component.

32. (Currently amended) A mapping tool with a graphical user interface for creating a mapping between a source object having a source object node and a target object having a target object node, comprising:

means for providing a function object having a script component with computer-executable instructions for performing a function, a graphical component associated with the function and having an input and an output, and an interface component;

means for displaying the graphical component in the graphical user interface;

means for displaying a source object and a target object in the graphical user interface, the source object comprising a plurality of hierarchically arranged source object nodes conforming to a first schema and the target object comprising a plurality of hierarchically arranged target object nodes conforming to a second schema;

means for graphically associating a source object node with the input and for graphically associating a target object node with the output; and

means for creating a mapping including the computer-executable instructions, the mapping operative to perform the function according to the source object node and to provide an output value associated with the target object node according to the function.

33. (Currently amended) In a mapping tool with a graphical user interface, a method of creating a function object for use in creating a mapping between a source object and a target object, the method comprising:

creating a script component having computer-executable instructions for performing a function using the user interface;

creating a graphical component associated with the function and having an input and an output, the input configured to be associated with at least one source object node

of the source object, the source object conforms to a first schema, and the output configured to be associated with at least one target object node of the target object, the target object conforms to a second schema;

creating an interface component that provides the script component to a compiler in the mapping tool and provides the graphical component to the graphical user interface; and

associating the script component, the graphical component, and the interface component.

34. (Previously presented) The method of claim 33, creating the script component further comprises receiving a user-defined text file including the computer-executable instructions and creating the script component using the computer-executable instructions from the text file.

35. (Previously presented) The method of claim 34, the computer-executable instructions include one of basic, visual basic, VB script, C++, visual C++, java, java script, and Perl .

36. (Previously presented) The method of claim 33, creating the script component comprises receiving function information from a user and creating the computer-executable instructions based on the function information.

37. (Previously presented) The method of claim 33, creating the interface component comprises:

receiving a text file from a user, the text file includes information related to the function; and

creating the interface component according to the information in the text file.

38. (Previously presented) The method of claim 37, the information related to the function is in XML.

39. (Original) The method of claim 33, further comprising:
prompting a user for information related to the function object;
receiving prompted information from the user via the graphical user interface;
creating the script component having computer-executable instructions for performing a function using the prompted information; and
creating the graphical component associated with the function and having an input and an output using the prompted information.

40. (Previously presented) The method of claim 39, prompting a user for information related to the function object comprises providing a wizard in the graphical user interface.

41. (Currently amended) A function object creation tool for creating a function object for use in creating a mapping in a mapping tool between a source object and a target object, the function object creation tool comprising:
means for creating a script component having computer-executable instructions for performing a function using the user interface;
means for creating a graphical component associated with the function and having an input and an output, the input configured to be associated with a source object node from amongst a hierarchical arrangement of source object nodes conforming to a first document structure and the output configured to be associated with a target object node from amongst a hierarchical arrangement of target object nodes conforming to a second document structure;
means for creating an interface component that provides the script component to a compiler in the mapping tool and provides the graphical component to the graphical user interface; and
means for associating the script component, the graphical component, and the interface component.

42. (Currently amended) A computer-readable medium having computer-executable instructions for performing the following steps:

- providing a function object having a script component with computer-executable instructions for performing a function, a graphical component associated with the function and having an input and an output, and an interface component;

- displaying the graphical component in a user interface;

- displaying a source object and a target object in the user interface, the source object displayed as a hierarchically arranged collection of source object nodes conforming to a first schema and the target object displayed as a hierarchically arranged collection of target object nodes conforming to a second schema;

- graphically associating a source object node with the input using a user interface selection device;

- graphically associating a target object node with the output using the user interface selection device; and

- creating a mapping including the computer-executable instructions, and operative to perform the function according to the source object node, and to provide an output value associated with the target object node according to the function.

43. (Currently amended) A computer-readable medium having computer-executable instructions for performing the following steps:

- creating a script component having computer-executable instructions for performing a function;

- creating a graphical component associated with the function and having an input and an output, the input configured to be associated with a source object node from amongst a hierarchical arrangement of source object nodes conforming to a first document structure and the output configured to be associated with a target object node from amongst a hierarchical arrangement of target object nodes conforming to a second document structure;

- creating an interface component that provides the script component to a compiler in a mapping tool and provides the graphical component to a graphical user interface; and

associating the script component, the graphical component, and the interface component.

44. (Currently amended) A function object for use in creating a mapping[[,]] between a source object having a source object node and a target object having a target object node, comprising:

- a script component having computer-executable instructions for performing a function;

- a graphical component associated with the function having an input and an output the graphical component configured to allow a user to graphically associate the input with a source object node and to associate the output with a target object node, the source object node selectable from a plurality of hierarchically arranged source object nodes conforming to a first schema and the target object node selectable from a plurality of hierarchically arranged target nodes conforming to a second schema; and

- an interface component having a globally unique identifier, the interface component provides the script component to a compiler and provides the graphical component to a graphical user interface.

45. (Currently amended) A computer-readable medium having computer-executable instructions for performing the following steps:

- creating a script component having computer-executable instructions for performing a function;

- creating a graphical component associated with the function and having an input and an output, the input configured to be associated with a source object node that is amongst a plurality of source object nodes, the source object node conforms to a first schema, and the output configured to be associated with a target object node that is amongst a plurality of target object nodes, the target object node conforms to a second schema;

- creating an interface component that provides the script component to a compiler and provides the graphical component to a graphical user interface; and

associating the script component, the graphical component, and the interface component.